



Università degli Studi di Bologna

Corso di Laurea in Ingegneria Informatica

Presentazione

Ingegneria del Software T
Anno accademico 2021/22

Prof. MARCO PATELLA

Dipartimento di Informatica – Scienza e Ingegneria (DISI)



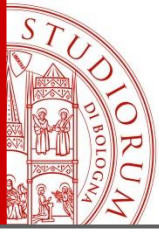
Il gruppo di lavoro

- Docente: Prof. Marco Patella
 - marco.patella@unibo.it
 - <http://www-db.disi.unibo.it/~mpatella/>
 - Ricevimento: gio 15-17 c/o palazzina DISI
- Tutor: ???



Una (prima) domanda fondamentale

- Che cosa significa ingegneria del software?



Risposta...

- È l'ingegneria del software



...e cos'è l'ingegneria?

- L'ingegneria è la disciplina, a forte connotazione tecnico-scientifica, che ha come obiettivo l'applicazione di conoscenze e risultati delle scienze matematiche, fisiche e naturali per produrre sistemi e soluzioni in grado di soddisfare esigenze tecniche e materiali della società attraverso le fasi della progettazione, realizzazione e gestione degli stessi.
- Applicando in questo senso le norme tecniche, fornisce metodologie, progetti e specifiche per la progettazione, realizzazione e gestione di un bene fisico, un prodotto o un servizio più o meno complesso, e più generalmente per lo sviluppo e il controllo di un processo industriale con un opportuno sistema.

(sorgente: wikipedia)



Una seconda domanda fondamentale

- Che cosa significa scrivere del buon software?



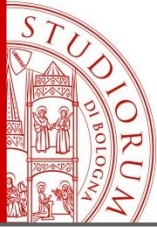
Alcune risposte tipiche

- **Programmatore C:**

“Scrivere del buon software significa
ottimizzare ogni istruzione, in modo da ottenere
il codice più compatto ed efficiente possibile”

- **Programmatore Visual Basic:**

“Scrivere del buon software significa
fornire le funzionalità richieste dall'utente
nel minor tempo possibile e con il minor costo possibile,
indipendentemente da come si arriva al risultato”



Risposta dell'ingegnere del software

Scrivere del buon software significa

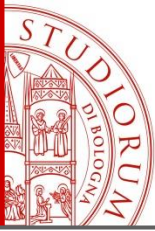
- Trovare il miglior equilibrio fra diversi fattori:
 - la **soddisfazione dell'utente**
 - l'**organizzazione** dell'applicazione
 - la **facilità di estensione** dell'applicazione
 - la **comprensibilità** delle soluzioni adottate
 - il **sapere valutare** e **scegliere** le giuste tecnologie e linguaggi
 - saper **tenere conto anche del fattore tempo**
 - ...
- Adottare tecniche adeguate a gestire la crescente complessità delle applicazioni



Risposta dell'ingegnere del software

Scrivere del buon software significa

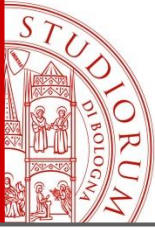
- Utilizzare al meglio l'investimento, spesso ingente, necessario per produrre un'applicazione, garantendo in particolare:
 - Il **maggior tempo di vita possibile**
 - Il **riutilizzo** in altri progetti di parte del codice prodotto



Ingegnere del Software

Compiti principali

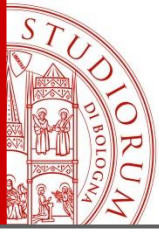
- Affrontare in modo sistematico e misurabile:
 - l'analisi dei requisiti
 - l'analisi del dominio
 - l'analisi del problema
 - la progettazione
 - la realizzazione
 - l'utilizzo
 - la manutenzionedei prodotti software
- Studiare le strategie per realizzare il punto precedente



Ingegneria del Software

Di cosa si occupa

- Gestione del **processo di sviluppo** del software:
 - Attività di **analisi**
 - Attività di **progettazione**
 - Attività di **codifica**
 - Attività di **verifica e convalida** (testing)
- Attività di **tipo gestionale**
 - Stime dei costi (e dei tempi)
 - Gestione dei progetti (delle persone, pianificazione)
 - Gestione dei rischi
 - Gestione della qualità
- Metriche



Obiettivi del corso

- Fornire i **concetti di base** dell'ingegneria del software
- Esaminare e utilizzare tecniche ***object-oriented*** per l'analisi, la progettazione e la realizzazione di applicazioni software



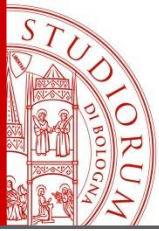
Obiettivi del corso

- Fornire nozioni avanzate sulla tecnologia .NET
 - *Framework .NET*
 - Linguaggio C#
 - Delegati, eventi
 - Attributi, introspezione (*reflection*)
 - *Garbage collector*

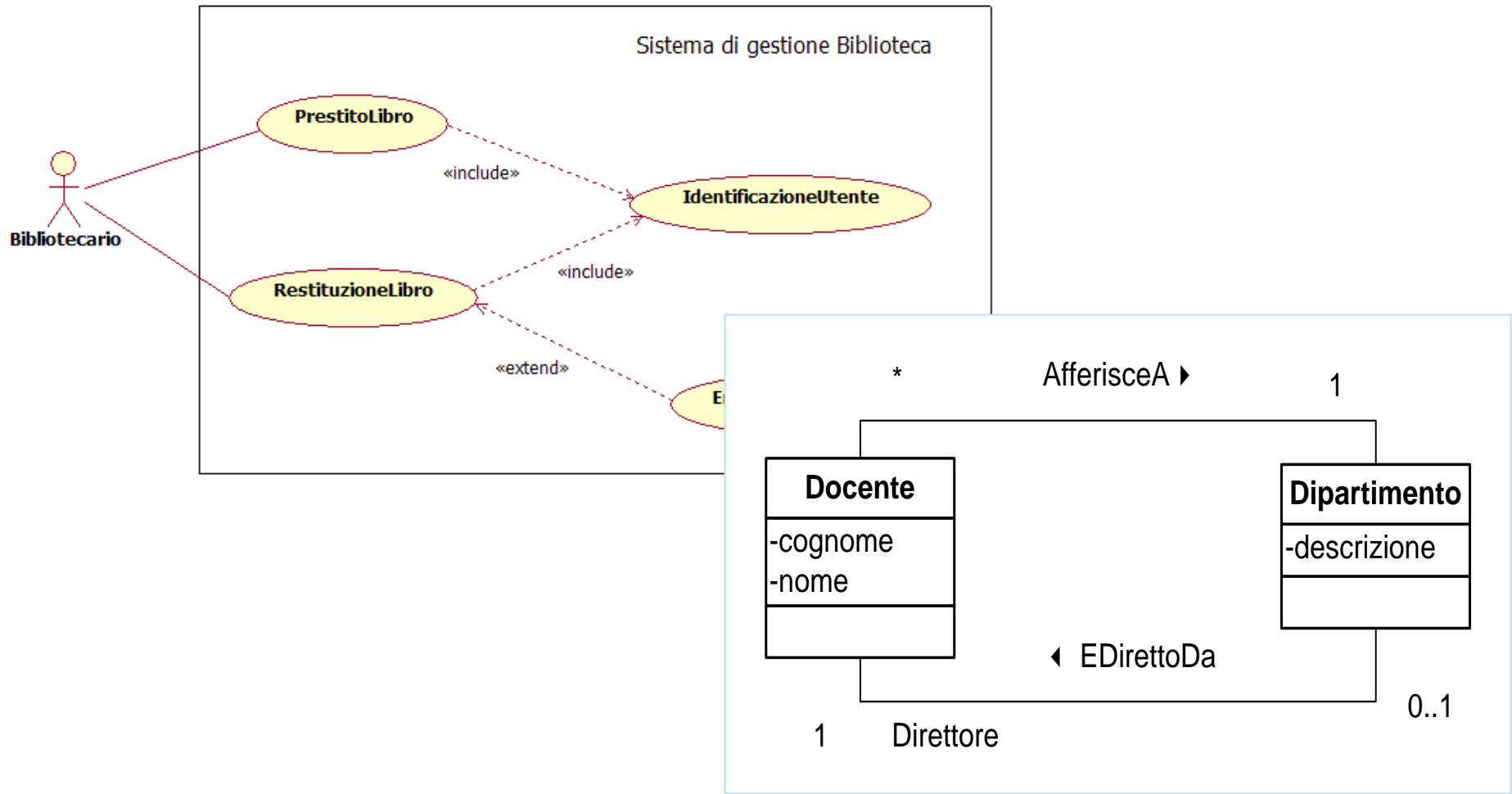


Linguaggio di modellazione

- Durante il processo di sviluppo del software è indispensabile poter utilizzare **un linguaggio per costruire i modelli** da discutere con il cliente e gli altri sviluppatori
- Un linguaggio di programmazione?
No, perché **un modello non è il codice!**
- Occorre una **forma descrittiva intermedia**, adeguata al livello descrittivo che serve...
- ... magari un linguaggio **visuale**
(una figura è meglio di mille parole)
- **UML** - *Unified Modeling Language*



Linguaggio di modellazione





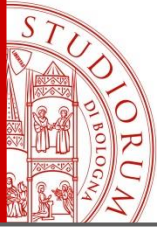
Design Pattern

- L'ingegneria tradizionale si avvale sempre **di soluzioni standard a problemi standard**
- Soluzioni validate dall'esperienza, che hanno dimostrato di **funzionare bene** in un gran numero di situazioni.
- Lo stesso approccio, seppur più recentemente perché è scienza più giovane, è stato adottato anche dall'ingegneria dell'informazione: in questo senso si parla di design pattern
- Un design pattern è *una soluzione progettuale generale a un problema ricorrente*.
- ... una descrizione o modello logico da applicare per la risoluzione di un problema che può presentarsi in diverse situazioni durante le fasi di analisi e progettazione del software



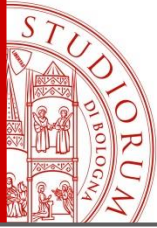
Design Pattern

- Durante la progettazione è indispensabile conoscere e utilizzare i **design pattern**, al fine di:
 - risolvere problemi progettuali specifici
 - rendere i progetti *object-oriented* più flessibili e riutilizzabili
- Ogni design pattern
 - cattura e formalizza l'**esperienza acquisita** nell'affrontare e risolvere uno specifico problema progettuale
 - permette di **riutilizzare** tale **esperienza** in altri casi simili



Testi consigliati

- Slide viste a lezione
 - <https://virtuale.unibo.it/course/view.php?id=27149>
- ***C. Larman, Applicare UML e i pattern***
Analisi e progettazione orientata agli oggetti
(quarta edizione), Pearson, 2016
- ***E. Gamma, R. Helm, R. Johnson, J. Vlissides, Design Patterns – Elements of Reusable Object-Oriented Software***, Addison Wesley, 1998



Bibliografia

- *M. Fowler*, **UML Distilled** (4a edizione italiana), Addison Wesley, 2010
- *S. Bennett, J. Skelton, K. Lunn*, **Introduzione a UML**, McGraw-Hill, 2002
- *W. Zuser, S. Biffl, T. Grechenig, M. Köhle*, **Ingegneria del software con UML e Unified Process**, McGraw-Hill, 2004
- *J. Arlow, I. Neustadt*, **UML e Unified Process** – Analisi e progettazione object-oriented, McGraw-Hill, 2003



Bibliografia

- *I. Sommerville*, **Ingegneria del software** (10a edizione), Pearson, 2017
- *R. S. Pressman*, **Principi di Ingegneria del Software** (quinta edizione), McGraw-Hill, 2008



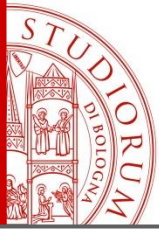
Orario delle lezioni

| | | |
|-----------|---------------|------------|
| Lunedì | 15:00 – 17:00 | Aula 2.3 |
| Mercoledì | 15:30 – 18:30 | Aula 2.4 |
| Giovedì | 15:00 – 18:00 | Aula Magna |



Esame

- Ingegneria del Software T (9 CFU)
 - Scritto (1/3) + Progetto&Discussione (2/3)
- Ingegneria del Software L-A (6 CFU)
 - Orale (1/2) + Progetto (1/2)
 - Orale (programma 2016/17)



Scritto

- **3 Domande – 1 ora di tempo**
 - a risposta aperta
 - su tutto il programma del corso
 - non è possibile utilizzare appunti o altro



Progetto

- **Progetto:**
 - Analisi dei requisiti
 - Analisi del problema
 - Progettazione dell'applicazione
 - Realizzazione di un prototipo
- **Da realizzarsi in gruppo (3 persone)**
 - La suddivisione del lavoro è orizzontale, non verticale
 - Ognuno partecipa a ogni fase del progetto



Progetto

- **Il tema del progetto *è a libera scelta***
 - Mandare *un abstract* al docente e tutor dove viene indicato chiaramente il dominio applicativo e le funzionalità (documento dei requisiti)
 - Attesa validazione dell'abstract
- **Invio del progetto a docente e tutor *20 giorni* prima della data indicata per la discussione**
 - Il tutor darà una prima valutazione del lavoro
 - Se il progetto supera una soglia minima di punteggio è possibile passare alla discussione con il docente, altrimenti bisognerà sistemare il lavoro



Discussione

- **Discussione del progetto con il docente**
 - Presentazione (ad es. ppt) + valutazione prototipo
 - Ciascun elemento del gruppo presenta una fase del progetto, ma tutti partecipano alla discussione
 - La valutazione è **personale**, non di gruppo



Date Scritto

| Data | Luogo | Orario | Turni |
|-------------|--------------|---------------|--------------|
| 17/06 | LAB4 | 10-13 | 2 |
| 01/07 | LAB3 | 10-13 | 2 |
| 15/07 | LAB9 | 10-13 | 2 |
| (15/09) | LAB4 | 10-13 | 2 |

Liste su almaesami



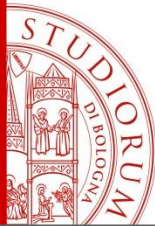
Date Progetto

- **Appelli di giugno/luglio**
 - Pubblicheremo le date verso aprile tenendo conto anche delle vostre richieste
- **Da settembre in poi**
 - Non ci saranno più appelli fissati, quando avete il progetto pronto ce lo mandate con una indicazione temporale (es. dal-al, non prima del/non oltre il) di quando vorreste fare la discussione



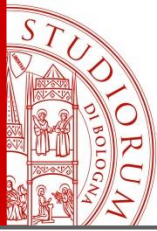
Esame - regole

- È possibile discutere il progetto anche se prima non si è superata la prova scritta
- Il voto conseguito in entrambe le prove viene mantenuto per i 6 appelli successivi, a meno di nuove consegne che vanno ad annullare il voto precedente
- Sul sito del corso troverete un file con tutte le regole di dettaglio



Software per diagrammi UML

- Microsoft Visio
 - Disponibile gratuitamente con MS Azure Dev Tools for Teaching
 - Non offre la possibilità di disegnare diagrammi E/R
- draw.io
 - Soluzione web gratuita, interoperabile con servizi di archiviazione in cloud (... o con il proprio disco rigido)
- [LucidChart](https://lucidchart.com)
 - Soluzione web
 - Offre nativamente opzioni di collaborazione
- DB-MAIN (solo per diagrammi E/R)



Software per lo sviluppo in C#

- Visual Studio
 - Disponibile per Windows e Mac
 - Versione “community”, meno funzioni, gratuita
 - Versione “enterprise” disponibile gratuitamente con MS Azure Dev Tools for Teaching, [confronto](#)
- [Visual Studio Code](#)
 - Disponibile gratuitamente per Windows, Mac, GNU/Linux
 - Meno funzionalità rispetto a VS (a metà tra un editor e un IDE completo)
- [Toolchain .NET Core da riga di comando](#)
 - Interoperabile con VS, disponibile per Windows, Mac, GNU/Linux
 - Offre strumenti per automazione dei test, CI/CD ecc.



Licenza Visio / VS - I

<https://azureforeducation.microsoft.com/devtools>



The screenshot shows the Microsoft Azure website. The top navigation bar includes the Microsoft Azure logo, 'Contact Sales', 'Search', 'My account', and 'Portal'. Below this is a secondary navigation bar with links for 'Overview', 'Solutions', 'Products' (highlighted with a dropdown arrow), 'Documentation', 'Pricing', 'Training', 'Marketplace', 'Partners', 'Support', 'Blog', and 'More'. A cookie consent banner is visible, stating 'We use cookies to improve your experience on our websites and for advertising.' with 'Accept all' and 'Manage cookies' buttons. The main content area is titled 'Azure Dev Tools for Teaching' and includes links for 'Enroll or Renew', 'Manage', and 'FAQ'. The text informs students that developer tools and learning resources previously part of their Imagine account are now available with Azure Dev Tools for Teaching. It instructs users to sign in using a Microsoft Account and provides a 'Sign In' button. A note mentions using the email from the previous Imagine subscription. A link to the 'help guide' and 'student FAQ' is also provided.

Microsoft Azure

Contact Sales Search My account Portal

Overview Solutions **Products** Documentation Pricing Training Marketplace Partners Support Blog More

We use cookies to improve your experience on our websites and for advertising. [Privacy Statement](#) **Accept all** **Manage cookies**

Azure Dev Tools for Teaching Enroll or Renew Manage FAQ All Microsoft  

Students—you're almost there! The developer tools and learning resources that were previously part of your Imagine account are now available with Azure Dev Tools for Teaching. Sign in using the button below—you'll be taken to a page requesting you to sign in using a Microsoft Account. Learn about Microsoft Accounts [here](#).

Note: Please use the email you provided for your previous Imagine subscription access when creating a new Microsoft Account.

Sign In

If you are having issues getting access, please reference our [help guide](#). For additional support, please check out [student FAQ](#).



Licenza Visio / VS - II

Effettuare il log-in con il proprio account istituzionale

Alma Mater Studiorum
University of Bologna



Enter the University institutional credentials.

Sign in

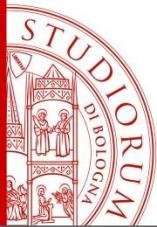
[Having trouble logging in?](#)

[Forgot your credentials?](#) | [Do you want to change your password?](#)

More information about credential

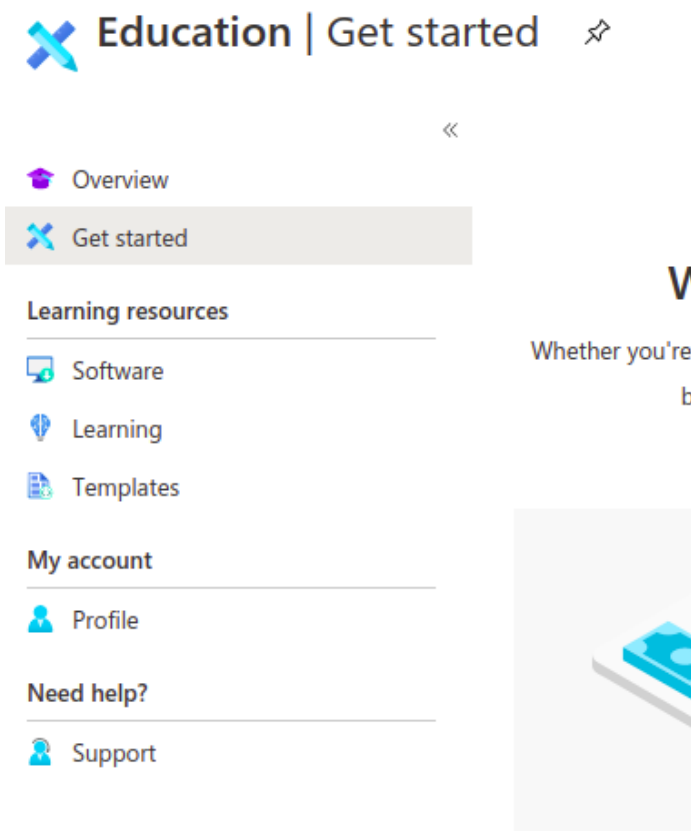
Choose the appropriate credential type:

- [@studio.unibo.it](#)
- [@unibo.it](#)
- [@esterni.unibo.it](#)



Licenza Visio / VS - III

Nella barra laterale a sinistra selezionare “Software”





Licenza Visio / VS - IV

Servendosi della barra di ricerca,
cercare il software desiderato

×



Product category : **All**

Operating System : **All**

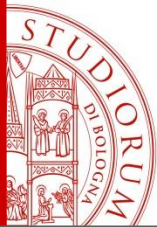
System type : **64 bit,Service**

Product language : **English,Multilanguage**

1 Item



| Name ↑↓ | Product category ↑↓ | Operating System ↑↓ | System type ↑↓ | Language |
|---|---------------------|---------------------|----------------|----------|
| Visio Professional 201... | Productivity Tools | Windows | 64 bit | Eng |



Licenza Visio / VS - V

Generare il link di download e la chiave di registrazione,
da inserire al primo avvio dopo l'installazione

Operating System

Windows

Product language

English

System

64 bit

View Key

Generate URL